SFC WG
Internet-Draft
Intended status: Standards Track
Expires: June 30, 2021

R. Chen
ZTE Corporation
X. Liu
Volta Networks
H. Chen
China Telecom
W. Wei
ZTE Corporation
T. Ao
individual

December 27, 2020

YANG data model for SFF
draft-ao-sfc-yang-03

Abstract

This document is to define the YANG data model for SFF configuration.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on June 30, 2021.

Copyright Notice

Copyright (c) 2020 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must

include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 2
2. Design tree for SFF YANG data model 2
3. YANG data model for SFF configuration 3
4. Security Considerations 10
5. IANA Considerations 10
6. References 10
6.1. Normative References 10
6.2. Information References 11
Authors' Addresses 11

1. Introduction

YANG[RFC6020]is a data definition language that was introduced to define the contents of a conceptual data store that allows networked devices to be managed using NETCONF [RFC6241]. This document defines a YANG data model for the configuration of SFF which data plane has been defined in [RFC8300].

2. Design tree for SFF YANG data model

```
module: ietf-sff
 +--rw sff
    +--rw sff* [sff-name]
       +--rw sfc-enable?   boolean
       +--rw sff-name                                sff-name
       +--rw ip-mgmt-address?                        inet:ip-address
       +--rw sff-locator* [name]
       |  +--rw sff-loacator-name                    sff-loacator-name
       |  +--rw (sff-transport-locator)?
       |  |  +--:(ipv4}
       |  |  |  +--rw remote-ipv4?      inet:ipv4-address
       |  |  +--:(ipv6)
       |  |  |  +--rw remote-ipv6?      inet:ipv6-address
       |  |  +--:(mac)
       |  |  +--rw remote-mac?       yang:mac-address
       |  |  +--:(vxlan-gpe)
       |  |  |  +--rw source-ip?        inet:ipv4-address
       |  |  |  +--rw destination-ip?   inet:ipv4-address
       |  |  |  +--rw vni               uint32
       |  |  +--:(mpls)
       |  |      +--rw mpls-label?       uint32
```

```
           |  +--rw transport?                identityref
           +--rw connected-sf*[name]
           |  +--rw sf-name                  sf-name
           |  +--rw sf-type                 sf-type
           |  +--rw sff-sf-locator*
           |  |  +--rw sf-locator-name        sf-locator-name
           |  |  +--rw sff-loacator-name      sff-locator-name
           |  +--rw sff-interfaces* [sff-interface]
           |     +--rw sff-interface           string
           +--rw connected-sff* [name]
              +--rw sff-name                 sff-name
              +--rw sff-sff-locator
              |  +--rw (locator-type)
              |  |   +--:(ipv4}
              |  |   |  +--rw remote-ipv4      inet:ipv4-address
              |  |   +--:(ipv6)
              |  |   |  +--rw remote-ipv6      inet:ipv6-address
              |  |   +--:(mac)
              |  |   +--rw remote-mac       yang:mac-address
              |  |   +--:(vxlan-gpe)
              |  |   |  +--rw source-ip?        inet:ipv4-address
              |  |   |  +--rw destination-ip?   inet:ipv4-address
              |  |   |  +--rw vni               uint32
              |  |   +--:(mpls)
              |  |      +--rw mpls-label?       uint32
              |  +--rw transport?                identityref
                         +--rw sff-interfaces* [sff-interface]
                             +--rw sff-interface           string
```

3. YANG data model for SFF configuration

This container defines a YANG model to configurate of SFF.The SF Type listed in this YANG model is referenced by [I-D.ietf-sfc-use-case-mobility] and [I-D.ietf-sfc-dc-use-cases].

```
<CODEBEGINS> file "ietf-sff@2020-12-10.yang"

module ietf-sff {
yang-version 1.1;

namespace "urn:ietf:params:xml:ns:yang:ietf-sff";
prefix "sff";

import ietf-inet-types {
prefix "inet";
        reference "RFC6991: Common YANG Data Types";
```

```
        }
    import ietf-routing-types {
        prefix "rt-types";
        reference "RFC8294:Common YANG Data Types for the Routing Area";
           }
        import ietf-yang-types {
        prefix "yang";
                reference "RFC6991: Common YANG Data Types";
        }
        import ietf-interfaces {
       prefix "if";
       reference "RFC8343: A YANG Data Model for Interface Management";
        }
        organization "IETF SFC Working Group";

        contact
       "WG Web:   <http://tools.ietf.org/wg/sfc/>
        WG List:  <mailto:sfc@ietf.org>
            WG Chair:Jim Guichard
             <mailto:james.n.guichard@huawei.com>
        WG Chair:Joel M. Halpern
             <mailto:jmh@joelhalpern.com>
        Editor: Ran Chen
                    <mailto:150387479@qq.com>
            Editor: Xufeng Liu
                    <mailto: xufeng.liu.ietf@gmail.com>
                Editor: Huanan Chen
                    <mailto: chenhua6@chinatelecom.com>
        Editor: Wei Wei
                    <mailto:wei.wei26@zte.com.cn>
                Editor: Ting Ao
            <mailto:ao.ting@zte.com.cn>
                         ";
        description
         "The YANG module defines a generic configuration model for SFF.
                  Copyright (c) 2019 IETF Trust and the persons
          identified as authors of the code.  All rights reserved.

          Redistribution and use in source and binary forms, with or
          without modification, is permitted pursuant to, and subject
          to the license terms contained in, the Simplified BSD License
          set forth in Section 4.c of the IETF Trust's Legal Provisions
          Relating to IETF Documents
          (https://trustee.ietf.org/license-info).
          This version of this YANG module is part of RFC XXXX; see
          the RFC itself for full legal notices.";

       revision "2020-01-20"{
```

```
            description "Initial revision.";
        reference "RFC XXXX: YANG Data Model for SFC Protocol.";
     }

        revision "2020-02-06"{
            description "01 revision.";
        reference "RFC XXXX: YANG Data Model for SFC Protocol.";
     }

         revision "2020-12-10"{
            description "02 revision.";
        reference "RFC XXXX: YANG Data Model for SFC Protocol.";
     }

    /* Typedef */
        typedef sff-name {
         type string;
         description "Service Function Forwarder Name";
     }

    typedef sff-locator-name {
         type string;
         description "Service Function Forwarder data-plane-locator name";
     }

        typedef sf-locator-name {
         type string;
         description
           "A unique name for SF data-plane-locator";
         }

    typedef sf-name {
       type string;
       description "Service Function Name";
     }

    typedef sf-type {
       type string;
       description "Service Function type Name";
     }

         /* Identities */
    identity transport-type{
       description
         "Base identity from which specific transport types are derived.";
     }
        identity locator-transport-type {
       base "transport-type";
```

```
       description
         "This identity is used as a base for all transport types";
     }
    identity ipv4 {
       base locator-transport-type;
       description
         "This identity represents IPv4 transport type.";
     }
    identity ipv6 {
       base locator-transport-type;
       description
         "This identity represents IPv6 transport-type.";
     }
        identity mac {
       base locator-transport-type;
       description
         "This identity represents sfp mac transport-type .";
     }
        identity vxlan-gpe {
       base locator-transport-type;
       description
         "This identity represents vxlan-gpe transport-type.";
     }
         identity mpls {
       base locator-transport-type;
       description
         "This identity represents mpls transport-type.";
     }


    /*grouping*/
         grouping locator-transport-type {
       description
         "This group presents configuration for the overlay data plane
locator.This could be VXLAN-GRE, MPLS,IP,MAC,etc";
        choice transport-type {
         mandatory true;

                 case ipv4 {
                   description
                     "The configuration for overlay data plane which
encapsulation type is ethernet&ipv4.";
            leaf ipv4 {
            type inet:ipv4-address ;
            description "Data-plane IPv4 address.";
           }
            description "The configuration for overlay data plane which
encapsulation type is ethernet&ipv4.";
```

```
          }

                  case ipv6{
```

```
                  description
                    "The configuration for overlay data plane which
encapsulation type is ethernet&ipv6.";
           leaf ipv6 {
           type inet:ipv6-address ;
                   description "IPv6 address.";
            }
          }

                 case mac{
                   description
                    "The configuration for overlay data plane which
encapsulation type is mac.";
           leaf mac {
             type yang:mac-address ;
             description "MAC address.";
             }
              }

         case vxlan-gpe-nexthop {
                   description
                     "The configuration for overlay data plane which
encapsulation type is vxlan-gpe.";
             leaf source-ip {
               description "The source IP address.";
               type inet:ipv4-address ;
             }
             leaf destination-ip {
               description "The destination address.";
               type inet:ipv4-address ;
             }
            leaf vni {
              type uint32;
              mandatory true;
              description "VNI value of the tunnel.";
              }
            }

           case mpls-nexhop{
             description"The configuration for overlay data plane which
encapsulation type is mpls.";
             uses rt-types:mpls-label-stack;
                         }
                         description "The collection of all possible data-
plane locators.";
               }
                 leaf transport {
                  type identityref {
```

```
                base locator-transport-type;
            }
                  description
            "The encapsulation used to carry NSH packets";
          }
```

```
       }


    container sff {
      description
        "A service function forwarder is responsible for delivering traffic
received from the SFC network forwarder to one or more connected service
functions via information carried in the SFC encapsulation.
         ";
        leaf sfc-enable {
            type boolean;
            default false ;
            description "Enable SFC." ;
            }
                list sff {
         key "sff-name";
         description
           "a list of all SFF configurations in the domain.";
                 leaf sff-name {
           type sff-name;
           description
                    "The unique name of this service function forwarder.";
          }

         leaf ip-mgmt-address {
           type inet:ip-address;
           description
             "The IP and port used to configure this
              service-function-forwarder";
         }

          list sff-locator{
            key "sff-loacator-name";
            leaf sff-loacator-name {
              type sff-loacator-name ;
               description "A list of all data-plane-locators of this SFF." ;
                }
            container sff-transport-locator{
              description
               "The overlay data plane locator used by this SFF. This could be
VXLAN-GRE,MPLS,MAC,etc";
             uses locator-transport-type;
           }
                   description
               "The list of sff data plane locator related informations.";
         }

                 list connected-sf {
```

```
          key "sf-name";
          leaf sf-name {
            type sf-name;
```

```
             description
                 "The name of the service function.";
                    }
                    leaf sf-type {
                          type sf-type;
              description
               "Service Function type names such as firewall, dpi,etc";
          }
                  container sff-sf-locator {
             description
               "SFF and SF data plane locators to use when sending packets from
this SFF to the associated SF";
             leaf sf-locator-name {
               type sf-locator-name;
                           description
                 "The SF data plane locator to use when sending packets to the
associated service function";
             }
                         leaf sff-locator-name {
               type sff-locator-name;
               description
                 "The SFF data plane locator to use when sending
                  packets to the associated service function.";
             }
                         }
                        list sff-interfaces {
             key "sff-interface";
             leaf sff-interface {
               type string;
               description
                 "An individual interface on the SFF connected to the SF";
             }
             description
               "A list of interfaces on the SFF which are connected to the SF";
           }
                  description
             "A list of all Service Functions attached to this SFF.";
         }

                 list connected-sff{
           key "sff-name";
           leaf sff-name {
             type sff-name;
             description
               "The name of the SFF connected to this SFF";
           }
           container sff-sff-locator {
             description
```

```
"The SFF uses this data plane locator when sending
 packets to the associated SFF";
```

```
            uses locator-transport-type;
          }
          list sff-interfaces {
            key "sff-interface";
            leaf sff-interface {
              type string;
              description
                "An individual SFF interface connected to this SFF";
            }
                    description
              "A list of SFF interfaces connected to this SFF";
            }
                   description
            "A list of all Service Function Forwarders connected to
             this SFF";
           }
                    }

    }
}
        <CODE ENDS>
```

4. Security Considerations

TBD.

5. IANA Considerations

TBD.

6. References

6.1. Normative References

[RFC7665] Halpern, J., Ed. and C. Pignataro, Ed., "Service Function Chaining (SFC) Architecture", RFC 7665, DOI 10.17487/RFC7665, October 2015, <https://www.rfc-editor.org/info/rfc7665>.

[RFC8300] Quinn, P., Ed., Elzur, U., Ed., and C. Pignataro, Ed., "Network Service Header (NSH)", RFC 8300, DOI 10.17487/RFC8300, January 2018, <https://www.rfc-editor.org/info/rfc8300>.

6.2. Information References

[I-D.ietf-sfc-dc-use-cases]
Kumar, S., Tufail, M., Majee, S., Captari, C., and S. Homma, "Service Function Chaining Use Cases In Data Centers", draft-ietf-sfc-dc-use-cases-06 (work in progress), February 2017.

[I-D.ietf-sfc-use-case-mobility]
Haeffner, W., Napper, J., Stiemerling, M., Lopez, D., and J. Uttaro, "Service Function Chaining Use Cases in Mobile Networks", draft-ietf-sfc-use-case-mobility-09 (work in progress), January 2019.

Authors' Addresses

Ran Chen
ZTE Corporation

Email: chen.ran@zte.com.cn

Xufeng Liu
Volta Networks

Email: xufeng.liu.ietf@gmail.com

Huanan Chen
China Telecom

Email: chenhua6@chinatelecom.com

Wei Wei
ZTE Corporation

Email: wei.wei26@zte.com.cn

Ting Ao
individual

Email: 18555817@qq.com